

Curb Detection Based on Elevation Maps from Dense Stereo

Florin Oniga, Sergiu Nedevschi,
Technical University of Cluj-Napoca, Romania
Florin.Oniga, Sergiu.Nedevschi@cs.utcluj.ro

Marc Michael Meinecke
Volkswagen A.G., Germany
marc-michael.meinecke@volkswagen.de

Abstract

A new approach for the detection of curbs (borders of relevant traffic isles, sidewalks, etc) is presented, in the context of urban driving assistance systems. The 3D data from dense stereo is transformed into a rectangular elevation map. The 3D resolution and uncertainty of the stereo sensor are evaluated and used in the detection process. Edge detection is applied on the elevation map in order to highlight height variations. Relevant lines are extracted using a Hough accumulator. Each relevant line is validated as a curb by applying a set of proposed conditions. The algorithm was evaluated in an urban scenario. It works in real time and provides robust detection of curbs.

1. Introduction

Processing 3D data from stereo (dense or sparse) is a challenging task. A robust approach can prove of great value for a variety of applications in urban driving assistance.

There are two main algorithm classes, depending on the space where processing is performed: disparity space-based and 3D space-based. Most of the existing algorithms try to compute the road/lane surface, and then use it to discriminate between road and obstacle points.

Disparity space-based algorithms are more popular because they work directly on the result of stereo reconstruction: the disparity map. The “v-disparity” [1] approach is well known and used to detect the road surface in a variety of applications [2]. Unfortunately, it is not a natural way to represent 3D data and has some drawbacks: it assumes that the road should occupy most of the image, and it is sensitive to roll angle changes. If the roll angle of the ego car changes from the initial calibration, than the road profile

becomes blurry and harder to detect on the “v-disparity” image.

3D space-based algorithms are mainly used for ego-pose estimation [3], [4], but also for lane and obstacle detection [5], [6].

In [3] the ego-pose is estimated with respect to the road plane. The road plane is fitted (not in real-time) by a RANSAC-approach to the whole set of dense 3D points (after filtering non-road points). A constant band around the road is used to select inliers and outliers, even though this is against the fact that the 3D uncertainty from stereo increases with the depth. The assumption that most of the 3D points are road points is made again (if the ego car is close to a sidewalk with more 3D points than the road, it is likely to fail).

A planar road surface is estimated from tracking in [4]. The method provides robust numerical results, but fails if occlusions (obstacles) are in front of the ego car. Lack of high-gradient road features also leads to failure (lane markings, borders, etc.).

Obstacles are detected as clusters of image edge points reconstructed in the 3D space ([5]). Road features as lane-markings can also be detected as obstacles. Later in [6] a 3D lane model is proposed and used for obstacle/road points’ separation. Again the method requires high-gradient road features (edges) to be present and uses a constant band to select road inliers and outliers. The band is large and most of the curbs are classified as road inliers.

The algorithm that will be presented in this paper manages to solve most of the drawbacks presented above. Its aim is to detect curbs having a height of at least 5 cm.

It takes as input dense 3D reconstructed points, overcoming the lack of road edge features. To achieve real-time processing, the 3D set of points is transformed into a digital elevation map. Curbs are detected based on local derivatives of the image map, so the shape of the road surface is irrelevant. The 3D uncertainty increasing with the depth is evaluated in

order to compute the maximum depth for robust curb detection.

First we will present the model for depth resolution and uncertainty of the stereo sensor. After that, the detection algorithm will be presented and evaluated.

2. The quadratic road surface model

2.1. The depth resolution from stereo

The set of 3D points from dense stereo will be transformed into a rectangular elevation map, each cell containing a height value. Due to the perspective projection, 3D points reconstructed by stereovision are sparser with the depth. This effect is more visible for the road and traffic isles surfaces, because they are quasi-parallel to the optical axis of the camera.

A 3D space compressed with the depth was used in [5] to achieve connectivity for obstacle points. We will use the opposite approach, by applying a morphological dilation of heights on the elevation map. The amplitude of dilation will be computed for each cell from the stereo geometry.

The effect of decreasing depth resolution for road points is shown in Figure 1. The distance between image-adjacent 3D road points increases with the depth.

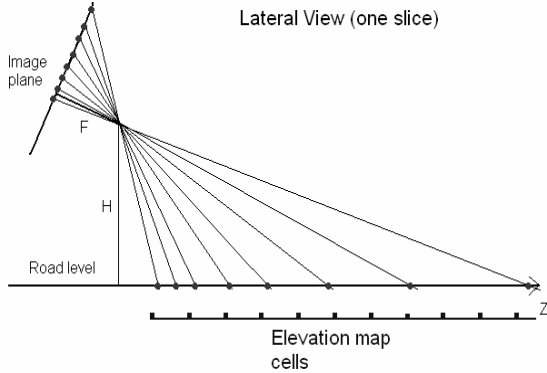


Figure 1. A lateral view of the depth resolution problem (F - focal length, H - height of the camera relative to the road level). Image-adjacent road points are not always adjacent in the elevation map.

Considering the canonical geometry of the stereo system and a planar road, the Z -value of a road point placed at the vertical coordinate Y_{im} (in pixels) in the image can be computed using (1). H is the height of the camera in the world reference frame, F the focal length of the stereo system, and α is the pitch angle of the camera.

$$Z(Y_{im}) = H * \frac{F * \tan(\frac{\pi}{2} - \alpha) + Y_{im}}{Y_{im} * \tan(\frac{\pi}{2} - \alpha) - F} \quad (1)$$

Furthermore the depth resolution ΔZ can be computed at any depth Z by (2).

$$\Delta Z = |Z(Y_{im}) - Z(Y_{im} + 1)| \quad (2)$$

Later in the paper it will be explained how to use this depth resolution model even for non-planar road surfaces.

2.2. The depth uncertainty from stereo

The 3D (localization) uncertainty is caused by a low accuracy computation of the disparity value and is mainly visible in the depth value. The height is also influenced and modeling the uncertainty will help for a robust detection of curbs.

In [7] a simple model for the uncertainty of the depth was proposed for a canonical stereo system. The depth uncertainty Z_{err} , for a point at depth Z , was modeled as a function (3) of the system's parameters (baseline B and focal F known from calibration) and of the disparity uncertainty D_{err} .

$$Z_{err} = \left| \frac{Z^2 * D_{err}}{B * F - Z * D_{err}} \right| \quad (3)$$

By using the standard canonical stereo equations, we extended this model to compute the uncertainty Y_{err} of a point with height Y and depth Z (4). H is the height of the camera in the world reference frame.

$$Y_{err} = \left| \frac{(Y - H) * Z_{err}}{Z} \right| \quad (4)$$

Figure 2 shows an example: the region for road inliers computed for a simple planar road surface assuming a disparity uncertainty D_{err} of 0.5 pixels. This value is quite common for surfaces with poor texture. At a depth of 10 meters, the height uncertainty is 3.5 cm, while at 35 meters it reaches 15 cm. Since we want to detect curbs at least 5 cm tall, we limited the depth of detection at 10 meters.

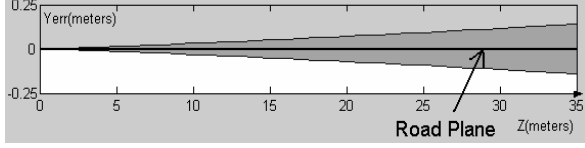


Figure 2. Lateral view of the inliers (darker grey) region around a planar road. The region is computed using the proposed model for the height uncertainty.

3. Algorithm description

The curb detection algorithm presented in this paper takes as input dense 3D reconstructed points. The output is a set of curbs, at most two (left/right). Each curb is represented as a chain of 3D points with associated height relative to the adjacent road surface. The borders of relevant traffic isles (elevated surfaces parallel to the road, such as sidewalks) are detected in this way.

To achieve real-time processing, the 3D set of points is transformed into a digital elevation map. All processing steps will be performed on the elevation map.

3.1. Building the elevation map

The elevation map representation is usually used for relief representation. An intensity map (image) is superimposed, from the top, to the 3D environment. Each pixel has a gray value proportional to the height of the underlying 3D location.

Computing the elevation map (Figure 3) from the set of dense 3D points is straightforward. A 3D space of interest (40m x 12m from bird-eye view in Figure 3) in front of the car is considered. The longitudinal Z and lateral X coordinates of each 3D point are scaled into the (image) coordinates space of the elevation map. The elevation map image has the same aspect ratio as the 3D space of interest. A cell has a size of 5 cm x 5 cm in the XZ world plane.

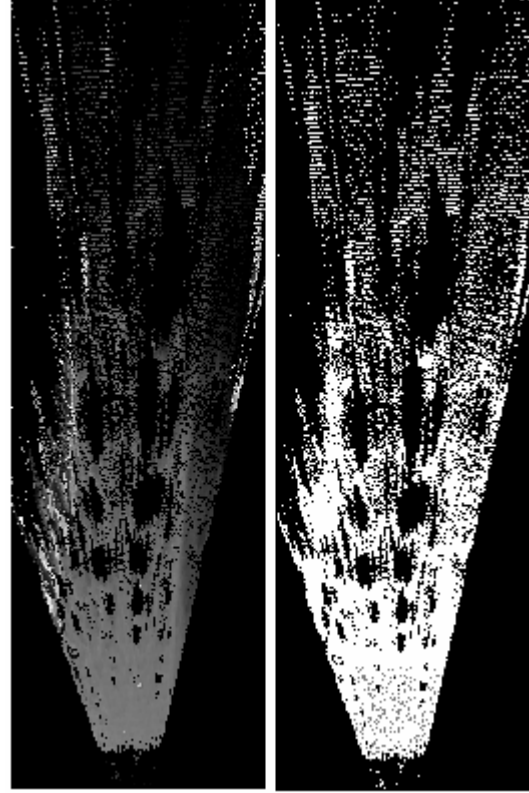
Each cell of the elevation map will contain the highest height of the 3D points contained in the cell. The height resolution of the elevation map about 4 mm. We centered the world zero level (road level at system calibration) at 128 in the elevation map for a better visualization. For all the images in this paper only heights around the road (1 m band) are displayed correctly due to a limited number of gray values (256). Internally the elevation map stores the whole range of heights on 32 bits, so no accuracy is lost.

3D points higher than 2 meters from the ground (zero level from calibration) will not be stored because

they are out of interest (it's highly unlikely that the ego car can have such a pitch angle variation that a curb at 10 meters depth will not be mapped into the elevation map). Empty cells are flagged and not used further.



a.



b.

c.

Figure 3. The elevation map (b) for the scene in a. In c valid cells (with 3D data) are highlighted with white.

As seen in Figure 3, the elevation map presents poor connectivity between road points at far depths. Since we want to perform derivative operations on the elevation map, a strong connectivity is required. The

amount of dilation needed for each cell is computed by dividing the depth resolution (2) with the size of one cell. To compensate for possible vertical road curvatures the amount of dilation obtained will be increased by 50%. This is necessary to avoid under-connectivity. Figure 4 shows the dilated elevation map.

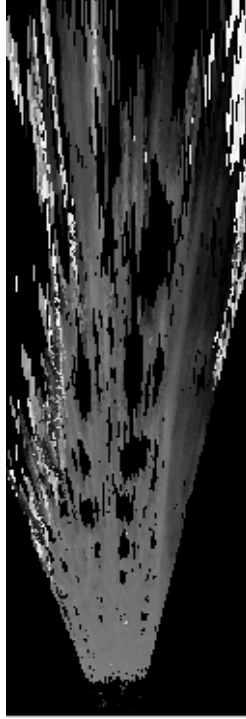


Figure 4. The dilated elevation map.

As a conclusion, instead of a set of 3D points without explicit connectivity, we obtained a discrete 3D space as an elevation map. Full connectivity of road (and surfaces close to road) points is available. For the sake of simplicity, in the next subsection (B) we will use the term *point* (described 3-dimensionally as the map coordinates and height) for a cell of the elevation map. All formulas presented in section II can be applied in the elevation map space (it is still a 3D space, but with discrete locations). As mentioned in the section II.B curbs will be detected only up to 10 meters depth on the elevation image.

3.2. Detecting curbs on the elevation map

The main goal was to develop a fast algorithm for detecting curbs nearby the ego car. Our approach involves the Hough transform, improved with a validation function to reject false positives.

The main steps of the process are:

1. Edge detection is applied on the elevation map (a region of interest), in order to emphasize height variations.
2. Relevant lines are computed from the set of edges, using the Hough accumulator.
3. Curbs are detected from the set of relevant lines by using a set of proposed criteria.

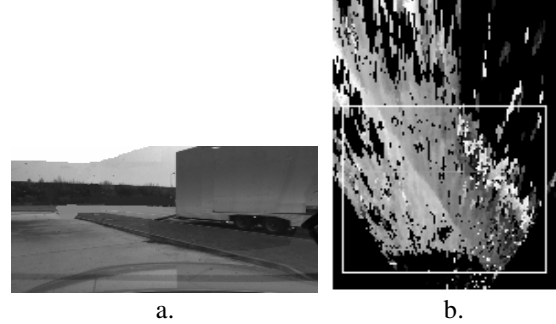


Figure 5. The elevation map (b) is computed for the scene in a. The region of interest used for curb detection is shown as a white rectangle (equivalent to 10x11 meters).

Curb points will present intensity changes on the elevation image, since they are points where the 3D height changes abruptly. It is obvious that curb points are included in the set of edge points of the elevation image. We used an optimal method [9] for detecting edges: the image gradient is computed with a Sobel filter, then non-maxima gradient points are suppressed, and finally edges are selected as points having the gradient above a threshold (Figure 6). Prior to detecting edges, a median 3x3 filter is applied to the elevation map to eliminate part of the spike-like noise (rough errors in the dense 3D reconstruction).

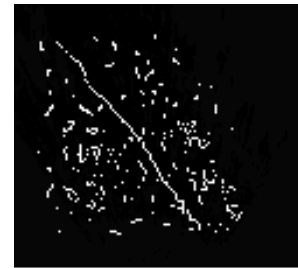


Figure 6. Edges detected on the elevation map from Figure 5.

The Hough accumulator [8] for lines is then built (Figure 7). Lines are represented in the (r, θ) space, which allows modeling a finite number of lines in the image (line parameters are bounded). The equation

used for representing a line is:

$$r = x * \cos(\theta) + y * \sin(\theta) \quad (5)$$

The Hough accumulator counts how many edge points are placed on each possible line in the image (Figure 7). We used a resolution of 1 degree for the angle θ and a resolution of 1 pixel for r .

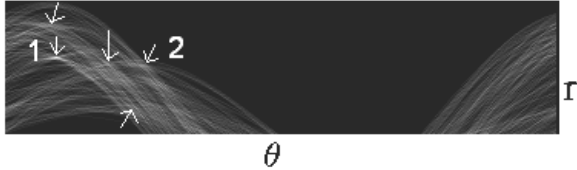


Figure 7. The Hough accumulator for the edges in Figure 6. Only the part with positive r is shown. The best five candidate lines are depicted by arrows.

The best five lines are selected as possible candidates and at most two will remain as final solution (left/right curbs). The process of selecting candidate lines from the Hough space consists of the following steps:

1. The line with the maximum score in the Hough accumulator is selected and stored.
2. Hough accumulator cells in the vicinity of the maximum score are nullified (the vicinity was chosen experimentally as a square of 15x15 cells). This is necessary to avoid multiple detection of the same 3D curb.
3. If less than five lines are selected, repeat the first two steps.

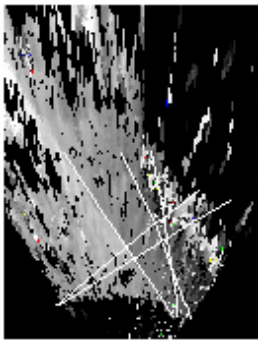


Figure 8. The best 5 lines selected from the Hough accumulator (Figure 7, depicted by arrows) are drawn on the elevation map.

The best five candidate lines are selected (Figure 8), based on their richness in edge points from the elevation image. As expected, due to noise in the elevation map image, most of the five lines are false

curbs. At most two lines will be marked as true curbs, by estimating the height variation profile of each line (a curb point should have specific variation of height in its vicinity).

Computing the height variation for each point of a line can be very unstable if only points adjacent to the line are used. A larger area must be used in order to have a result less influenced by noise. The shape of the area cannot be rectangular since it would not provide symmetry for all possible line orientations. Therefore, we used a circular mask (7x7 pixels, as in Figure 9.a) to estimate the height variation for each line point. The height variation for one point is computed as a difference function between points placed on the same side as the ego car (relative to the line) and the opposite points inside the circular nucleus.

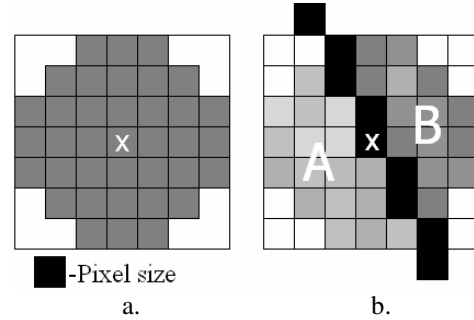


Figure 9. The circular mask (7x7 pixels, in a) used for computing the height variation for the central point x . An example with real data is shown in b, line pixels are drawn with black.

More precisely, an example with real data is presented in Figure 9.b: pixels inside the mask with their values, the line pixels drawn with black and, in the center, the current line point x for which the computation is performed. The following steps are applied to compute the height variation (for the center of the mask) along the normal of the line:

1. Median filtering is applied for each of the regions A (A is on the same side as the ego car) and B: pixels are sorted in ascending order in each region, and the middle values H_m of the ordered sets are selected as the heights for each region.
2. The height variation for the line point x is computed as $H_m(A) - H_m(B)$.

Finally, the height variation profile of each of the five candidate lines is evaluated by computing the height variation for each line point (Figure 10). A line point is considered curb point if it has a height between 5 and 35 centimeters. A score is computed for each

line, as the percentage of curb points of the total number of line points. A line is considered a curb if it has a score higher than 40%. The best two (or one, or none) curbs are selected, having the highest scores.

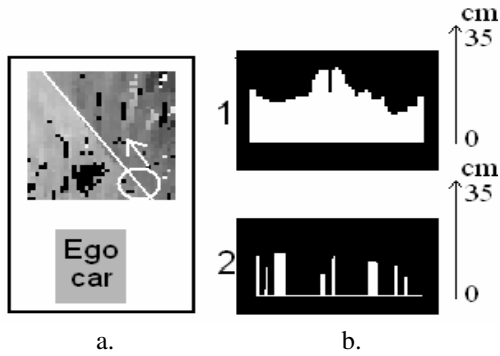


Figure 10. The process of computing the height variation profile of a line is performed by sliding a circular mask in a. In b the profile is drawn for lines marked as 1 and 2 in the Hough space from Figure 7. Only line 1 is classified as curb.

4. Results

The algorithm was implemented in C++. The dense 3D information was generated using a calibrated stereo rig with grayscale cameras and a commercial dense stereo board [10].

Due to the use of software-specific C optimizations and the elevation map representation, a processing time of 10 to 12 ms was achieved for the algorithm itself (on Pentium 4 at 2.6 Ghz). Building the elevation map is the most time consuming part, taking between 6 to 8 ms. The detection of curbs on the elevation map is very fast, about 4 ms.

Overall, with the image acquisition and the dense hardware reconstruction, a sustained processing frame-rate of 35 fps is obtained.

Regarding the robustness of the algorithm we performed the following evaluation that will be extended in the future:

- A number of 50 stereo images of different (random) scenes were selected (out of several hours of stored stereo images, recorded while driving the ego-car).
- Results were analyzed for each frame in terms of missed curbs and false curbs. Visual validation was performed by checking the projection of each 3D curb back onto the image plane.

For curbs higher than 5 cm the algorithm proved very robust, with no missing curbs (80 curbs were

present in the 50 scenes, 3 of the scenes are shown in Figure 13). On one of the 50 scenes, a false curb appeared due to poor 3D reconstruction (Figure 12).

Even though the proposed method was not designed for small curbs, it was also tuned and tested on smaller curbs (3-4 cm) but it is less robust: about 15% of the detected curbs were false ones. This is because small curbs have the height too close to the uncertainty of the stereo reconstruction. Future improvements of the stereo board used will improve the behavior for small curbs.

The algorithm also works for curved curbs, assuming the curvature is low (Figure 13.c).

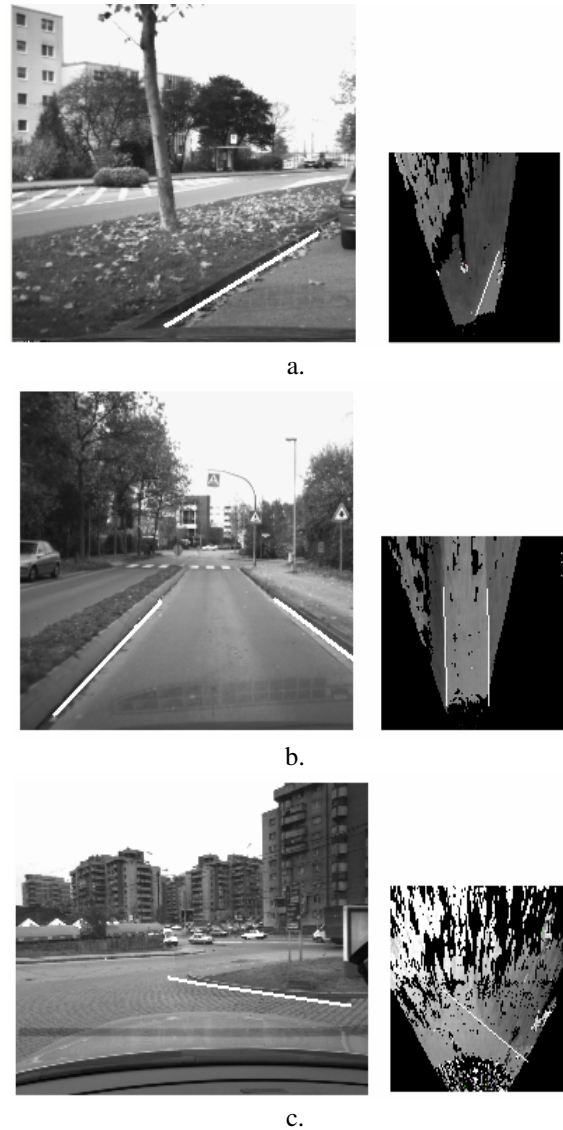


Figure 13. Results for various scenes, curbs are drawn on the elevation map and projected back onto the left image.

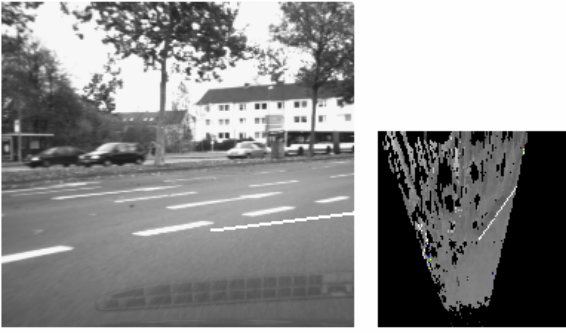


Figure 12. A false curb is detected due to a noisy 3D reconstruction (images are taken while the ego car was steering, so the lateral motion blur influenced the reconstruction).

5. Conclusions

A new curb detection algorithm was presented. The algorithm proved to be robust. Most of the erroneous detections are caused by bad 3D data from the dense stereo board. The current version copes well with most (but not all) types of errors in the 3D data set.

The proposed algorithm works in real-time and provides robust results. It can be used in a variety of applications that require detection of road-elevated delimiters such as curbs.

Future development and evaluation will be performed:

- Further and more complex evaluation of the proposed algorithm,
- Use the intensity image information (texture) to improve the sensitivity of the algorithm for small curbs (3-4 cm),
- The use of a neural network-based classifier to discriminate between false and true curbs, based on their height variation profile,
- Using tracking can greatly reduce the rate of false positives.

6. References

- [1] R. Labayrade, D. Aubert, and J.-P. Tarel, "Real time obstacle detection on non flat road geometry through V-disparity representation," in *IEEE Intelligent Vehicles Symposium*, Versailles, June 2002, pp. 646-651.
- [2] Broggi, C. Caraffi, P. Paolo Porta, and P. Zani, "The Single Frame Stereo Vision System for Reliable Obstacle Detection used during the 2005 DARPA Grand Challenge on TerraMax", in *IEEE Intelligent Transportation Systems Conference*, Toronto, Canada, September 17-20, 2006, pp. 745-752.
- [3] Sappa, D. Gerónimo, F. Dornaika, and A. López, "Real Time Vehicle Pose Using On-Board Stereo Vision System", *Int. Conf. on Image Analysis and Recognition*, LNCS Vol. 4142, Springer Verlag, Póvoa de Varzim, Portugal, September 18-20, 2006, pp. 205-216.
- [4] M. Cech, W. Niem, S. Abraham, and C. Stiller, "Dynamic ego-pose estimation for driver assistance in urban environments". In *IEEE Intelligent Vehicles Symposium*, Parma, Italy, 2004, pp. 43-48.
- [5] S. Nedevschi, R. Danescu, D. Frentiu, T. Marita, F. Oniga, C. Pocol, R. Schmidt, T. Graf, "High Accuracy Stereo Vision System for Far Distance Obstacle Detection", *IEEE Intelligent Vehicles Symposium (IV2004)*, Parma, Italy, pp. 292-297, 2004.
- [6] S. Nedevschi, R. Schmidt, T. Graf, R. Danescu, D. Frentiu, T. Marita, F. Oniga, C. Pocol, "3D Lane Detection System Based on Stereovision", *IEEE Intelligent Transportation Systems Conference (ITSC)*, Washington, USA, pp.161-166, 2004.
- [7] S. Nedevschi, F. Oniga, R. Danescu, T. Graf, R. Schmidt, "Increased Accuracy Stereo Approach for 3D Lane Detection", *IEEE Intelligent Vehicles Symposium*, (IV2006), June 13-15, 2006, Tokyo, Japan, pp. 42-49.
- [8] J. Canny, "A computational approach to edge detection", *IEEE Trans. Pattern Analysis and Machine Intelligence*, pp. 679-698, Nov. 1983.
- [9] D. H. Ballard, "Generalizing the Hough Transform to Detect Arbitrary Shapes", *Pattern Recognition*, 13(2):111-122, 1981.
- [10] J. Woodfill, G. Gordon, D. Jurasek, T. Brown, and R. Buck, "The Tyzx DeepSea G2 Vision System, A Taskable, Embedded Stereo Camera", *IEEE Conference on Computer Vision and Pattern Recognition*, (New York, NY), June 2006.